

We claim:

1. A system for processing data packets comprising a plurality of data blocks, the system comprising:

5 a controller; and

a plurality of data processing blocks connected in a sequence, each data processing block also connected to the controller for receiving respective configuration parameters from the controller, the configuration parameters being used for controlling the processing of the data blocks within each respective data processing block, wherein

10 each data processing block transmits a first data signal along with a first output data block associated with a particular data packet to a subsequent data processing block in the sequence, wherein the first data signal is propagated through the sequence of data processing blocks with the first output data block, and each data processing block changes to new configuration parameters upon receipt of the first data signal from a previous data processing block in the sequence, whereby the configuration parameters are only changed on a data packet boundary.

2. The system of claim 1, wherein each data processing block receives an input data block and provides an output data block utilizing a handshaking protocol using a ready to receive signal (RTR) to signify that the data processing block is ready to receive a data block and a ready to send signal (RTS) to signify that the data processing block is ready to send the data block, and a data block is transferred between a pair of data processing blocks only when both the RTR and RTS signals are set between the data processing blocks.

3. The system of claim 1, wherein the controller provides to a first data processing block in the sequence a first signal representative of the number of data blocks within a data packet.

4. The system of claim 3, wherein the first data processing block identifies a first data block in a subsequent data packet in response to the first signal.

5        5.        The system of claim 4, wherein the first data processing block provides a second signal to the controller in response to processing a last data block in the data packet, the second signal indicating that the data packet has been completely processed by the first data processing block and a new data packet can be loaded into the first data processing block.

10        6.        The system of claim 5, wherein a last data processing block in the sequence provides to the controller a third signal in response to receiving the first data signal from a previous data processing block, the third signal indicating that a first data blocks of the data packet has been processed by all of the data processing blocks in the sequence and that the controller can update the configuration parameters in the data processing blocks.

15        7.        The system of claim 6, wherein the controller provides to the first data processing block a fourth signal to indicate that processing of the data packet may start in response to the presence of the second and third signals.

20        8        The system of claim 1, wherein one of the data processing blocks receives the first data signal and an input data block, processes the input data block and provides a plurality of output data blocks, and provides the first data signal only with the first one of the output data blocks to a subsequent data processing block.

25        9.        In a system comprising a plurality of data processing blocks connected in a sequence, each data processing block also connected to a controller for receiving respective configuration parameters from the controller, a method for processing data packets comprising the steps of:

identifying, in each data processing block, a first data block associated with a particular data packet;

30        reading new configuration parameters from the controller in response to the identifying step;

performing data processing utilizing the configuration parameters read from the controller; and

transmitting a first data signal along with a first output data block associated with the particular data packet whereby the configuration parameters are changed in the data processing blocks at a data packet boundary.

5           10.    The method according to claim 9, further comprising the steps of providing a ready to send signal line and a ready to receive signal line between each one of the data processing blocks, and transmitting data blocks between the data processing block only when the ready to send signal and ready to receive signal are present.

10

          11.    The method according to claim 9, further comprising the step of providing, by the controller, to a first data processing block in the sequence a first signal representative of the number of data blocks within a data packet, wherein the first data processing block identifies a first data block in a subsequent data packet  
15   in response to the first signal.

15

          12.    The method according to claim 11, further comprising the step of providing, by the first data processing block in the sequence, to the controller a second signal in response to processing a last data block in the data packet, the  
20   second signal indicating that the data packet has been completely processed by the first data processing block and a new data packet can be loaded into the first data processing block.

20

          13.    The method according to claim 12, further comprising the step of  
25   providing, by a last data processing block in the sequence, to the controller a third signal in response to the receipt of the first data signal by the last data processing block, the third signal indicating that a first data block of the data packet has been processed by all of the data processing blocks in the sequence, the controller updating the configuration parameters in response to receipt of the third  
30   signal.

25

30

          14.    The method according to claim 13, further comprising the step of

providing, by the controller, to the first data processing block a fourth signal in response to the receipt of the second and third signals, the fourth signal indicating that the first data processing block may start processing a next data packet.